

of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-210097

(P2001-210097A)

(43) 公開日 平成13年8月3日(2001.8.3)

(51) Int.Cl.⁷

G 1 1 C 29/00

識別記号

6 5 5

F I

G 1 1 C 29/00

テーマコード(参考)

6 5 5 Z 5 L 1 0 6

6 5 5 S

審査請求 未請求 請求項の数 7 O L (全 14 頁)

(21) 出願番号

特願2000-18783(P2000-18783)

(22) 出願日

平成12年1月27日(2000.1.27)

(71) 出願人 000003078

株式会社京芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 伊 藤 治

神奈川県川崎市幸区小向京芝町1番地 株

式会社京芝マイクロエレクトロニクスセン

ター内

(74) 代理人 100064285

弁理士 佐藤 一雄 (外3名)

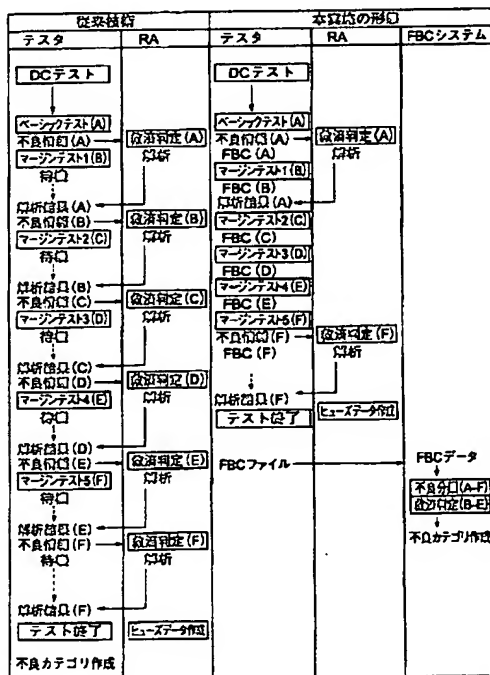
Fターム(参考) 5L106 DD25

(54) 【発明の名称】 半導体記憶装置の不良救済判定方法及びその装置

(57) 【要約】

【課題】 半導体記憶装置の不良判定処理の効率向上、及び不良箇所の特定を可能とする。

【解決手段】 テスタを用いて半導体記憶装置に試験項目毎に試験を行い、所定の大きさを有するセグメント毎の不良ビット数を測定して不良情報を作成し、テストから不良救済判定装置に不良情報を送信し、テストやヒューズデータ生成用不良解析装置とは独立して設けた不良救済判定装置を用いて、不良情報に基づいて、複数の不良レベルに存在するそれぞれの不良数を検出して不良レベルの分類を行い、各々の不良レベルの不良数とスペア数とを比較して、不良レベル毎に不良救済の可否を判断する。このように、不良解析及び不良救済判定処理をテスト及び不良解析装置から独立して行うことで、処理効率を向上させることができ、また不良救済不可となった不良レベルの特定が可能である。



1

【特許請求の範囲】

【請求項 1】冗長回路を有する半導体記憶装置に対して複数の試験項目毎に試験を行い、不良救済の可否を判定する方法において、

テストを用いて前記半導体記憶装置に試験項目毎に試験を行い、所定の大きさを有するセグメント毎の不良ビット数を測定して不良情報を作成する工程と、
前記テストから不良救済判定装置に前記不良情報を送信する工程と、

前記不良救済判定装置を用いて、前記不良情報に基づき、複数の不良レベルに存在するそれぞれの不良数を検出して不良レベルの分類を行う工程と、

前記不良救済判定装置を用いて、各々の不良レベルの不良数と、前記冗長回路に置換可能なスベア数とを比較して、不良レベル毎に不良救済の可否を判断する工程と、を備えたことを特徴とする半導体記憶装置の不良救済判定方法。

【請求項 2】前記不良救済判定装置とは独立した不良解析装置を用いて、前記不良情報に基づき不良箇所を前記冗長回路に置き換えるアドレス情報を生成する工程をさらに備えることを特徴とする請求項 1 記載の半導体記憶装置の不良救済判定方法。

【請求項 3】前記試験項目のうち所定のものに対しては、前記テストがこの試験項目の試験を行って作成した不良情報を前記不良解析装置に送信し、この試験項目における不良救済の可否を判断し、不良救済が不可であると判断した場合この段階で試験を終了する工程をさらに備えることを特徴とする請求項 2 記載の半導体記憶装置の不良救済判定方法。

【請求項 4】前記不良レベルの分類を行う工程は、前記半導体記憶装置の構造に応じて設定された n (n は 2 以上の整数) 個の不良レベル毎に不良数を検出するものであって、

第 1 の不良レベルの判定基準と、前記第 1 の不良レベルにおける判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、

前記第 1 の不良レベルよりも下位の第 2 の不良レベルの判定基準と、前記第 2 の不良レベルの判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、

...

第 $n-2$ の不良レベルよりも下位の第 $n-1$ の不良レベルの判定基準と、前記第 $n-1$ の不良レベルの判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、

第 $n-1$ の不良レベルよりも下位の第 n の不良レベルのセグメント内に存在する不良ビット数を、この第 n の不

2

良レベルの不良数とすることを特徴とする請求項 1 記載の半導体記憶装置の不良救済判定方法。

【請求項 5】前記不良救済の可否を判断する工程は、 n 個の不良レベル毎に不良救済の可否を判断するものであり、

前記第 1 の不良レベルの不良数と、この第 1 の不良レベルの救済処理単位内における置換可能なスベア数とを比較し、前記不良数が前記スベア数を超過している場合は救済不可と判断し、前記不良数が前記スベア数以下である場合は第 1 の不良レベルで救済可と判断し、

前記第 1 の不良レベルで救済可と判断した場合、前記第 2 の不良レベルの不良数と、この第 2 の不良レベルの救済処理単位内における置換可能なスベア数とを比較し、前記不良数が前記スベア数を超過している場合は救済不可と判断し、前記不良数が前記スベア数以下である場合は第 2 の不良レベルで救済可と判断し、

...

前記第 $n-2$ の不良レベルで救済可と判断した場合、前記第 $n-1$ の不良レベルの不良数と、この第 $n-1$ の不良レベルの救済処理単位内における置換可能なスベア数とを比較し、前記不良数が前記スベア数を超過している場合は救済不可と判断し、前記不良数が前記スベア数以下である場合は第 $n-1$ の不良レベルで救済可と判断し、前記第 $n-1$ の不良レベルの不良数と、この第 n の不良レベルの救済処理単位内における置換可能なスベア数とを比較し、前記不良数が前記スベア数を超過している場合は救済不可と判断し、前記不良数が前記スベア数以下である場合は第 n の不良レベルで救済可と判断することを特徴とする請求項 4 記載の半導体記憶装置の不良救済判定方法。

【請求項 6】冗長回路を有する半導体記憶装置に対して複数の試験項目毎に試験を行い、不良救済の可否を判定する装置において、

前記半導体記憶装置に試験項目毎に試験を行い、所定の大きさを有するセグメント毎の不良ビット数を測定して不良情報を作成するテストと、

前記テストから不良救済判定装置に前記不良情報を与えられ、前記不良情報に基づき、複数の不良レベルに存在するそれぞれの不良数を検出して不良レベルの分類を行い、各々の不良レベルの不良数と、前記冗長回路に置換可能なスベア数とを比較して、不良レベル毎に不良救済の可否を判断する不良救済判定装置と、を備えたことを特徴とする半導体記憶装置の不良救済判定装置。

【請求項 7】前記不良救済判定装置とは独立して、前記不良情報に基づき不良箇所を前記冗長回路に置き換えるアドレス情報を生成する不良解析装置をさらに備えることを特徴とする請求項 6 記載の半導体記憶装置の不良救済判定装置。

【発明の詳細な説明】

【0001】

3

【発明の属する技術分野】本発明は、半導体記憶装置の不良救済判定方法及びその装置に関する。

【0002】

【従来の技術】半導体記憶装置の試験を行う場合、半導体試験装置（以下、テストという）を用いた試験行程と、得られたデータに基づいて不良解析装置（Redundancy Analyzer、以下、RAという）を用いて救済可能か否かの判断を行う解析工程とを、従来は図1に示されたように並行して行っていた。

【0003】まず、テストを用いてDCテストを行う。さらに、例えば読み書き等を調べるベーシックテスト（A）を行い、不良情報（A）を生成する。

【0004】この不良情報（A）はRAに送信されて解析され、救済判定が行われる。この救済判定を行っている間、テストによりマージンテスト（B）が行われる。マージンテストは、試験項目（B）、（C）、…、

（F）毎に行われていく。救済可又は不可を示す解析結果（A）が得られると、RAからテストへこの情報が送信される。マージンテスト（B）による不良情報（B）がテストからRAに送信され、救済判定が行われる。

【0005】RAが救済判定を行っている間、テストがマージンテスト（C）を行う。RAが救済判定（B）を終えて解析結果（B）を得ると、RAからテストへこの情報が送信される。マージンテスト（C）による不良情報（C）がRAに送信され、救済判定が行われる。

【0006】以上の手順で、テストがマージンテスト（F）まで行い、並行してRAが救済判定（F）まで行って解析結果（F）を生成する。この解析結果（F）は、救済判定（A）～（F）までの全ての解析結果が累積された情報となっている。そこで、この解析結果

（F）を用いて、RAが不良箇所を冗長回路に置き換えるアドレスを生成するためのヒューズ回路における各ヒューズの溶断の有無を示すヒューズデータを生成して、試験が終了する。

【0007】ここで、救済判定（A）～（F）のうち、いずれかにおいて救済不可となるとその段階で試験が終了する。

【0008】

【発明が解決しようとする課題】しかし、従来の不良救済判定方法には次のような問題があった。

【0009】上述したように、テストによるベーシックテスト（A）及び各試験項目毎のマージンテスト（B）～（F）と、RAによる各試験毎の救済可否の判断とを、並行して行っていた。RAにより得られた解析結果（A）～（F）は、不良解析の為のデータとして、蓄積していた。

【0010】一般に、RAにより行われる救済判定処理にかかる時間は、半導体記憶装置の容量や救済アルゴリズムの複雑さ、さらにはRAの処理能力に依存する。従って、近年の半導体記憶装置の大容量化や救済アルゴリ

4

ズムの複雑化に伴い、救済判定処理時間が増大してきた。

【0011】この結果、テストがマージンテストを行うために要する時間よりも、RAがテスト項目毎に救済判定処理を行う時間の方が長くなってきた。このため、テストがマージンテスト（B）～（F）をそれぞれ終えた後、RAが救済判定（A）～（F）を終えるまでの間、テストが待機しなければならなかった。

【0012】テストは一般に高価であり、待機する時間を極力なくしたいという要請がある。待機時間をなくするためには、RAの処理速度を高速化すればよい。しかし、既に量産工場に導入している既存のRAを全て高速化するとすると、コストの大幅な増大は避けられない。

【0013】また、テストからRAに送信する不良情報（A）～（F）は、圧縮処理がなされていないのでデータ量が膨大である。このため、送信に時間がかかるという問題もあった。

【0014】さらに、不良解析を行う上での処理効率という観点から見た場合、上記方法により採取した情報では、救済不可と判定した試験項目を特定することは容易である。しかし、救済不可となった試験項目において、ブロック、ロー又はカラム、ビット等を単位としていずれの箇所が不良であったかという不良箇所を特定するための詳細な情報を得ることは不可能であった。

【0015】本発明は上記事情に鑑み、半導体記憶装置の試験を行う際の不良判定処理効率の向上及び不良箇所の特定を可能にする半導体記憶装置の不良判定処理方法及びその装置を提供することを目的とする。

【0016】

【課題を解決するための手段】本発明の半導体記憶装置の不良判定処理方法は、冗長回路を有する半導体記憶装置に対して複数の試験項目毎に試験を行い、不良救済の可否を判定する方法であって、テストを用いて前記半導体記憶装置に試験項目毎に試験を行い、所定の大きさを有するセグメント毎の不良ビット数を測定して不良情報を作成する工程と、前記テストから不良救済判定装置に前記不良情報を送信する工程と、前記不良救済判定装置を用いて、前記不良情報に基づき、複数の不良レベルに存在するそれぞれの不良数を検出して不良レベルの分類を行う工程と、前記不良救済判定装置を用いて、各々の不良レベルの不良数と、前記冗長回路に置換可能なスベア数とを比較して、不良レベル毎に不良救済の可否を判断する工程とを備えたことを特徴としている。

【0017】ここで、前記不良救済判定装置とは独立した不良解析装置を用いて、前記不良情報に基づき不良箇所を前記冗長回路に置き換えるアドレス情報を生成する工程をさらに備えることもできる。

【0018】また、前記試験項目のうち所定のものに対しては、前記テストがこの試験項目の試験を行って作成した不良情報を前記不良解析装置に送信し、この試験項

5

目における不良救済の可否を判断し、不良救済が不可であると判断した場合この段階で試験を終了する工程をさらに備えることもできる。

【0019】前記不良レベルの分類を行う工程が、前記半導体記憶装置の構造に応じて設定された n 個の不良レベル毎に不良数を検出するものであって、第1の不良レベルの判定基準と、前記第1の不良レベルにおける判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、前記第1の不良レベルよりも下位の第2の不良レベルの判定基準と、前記第2の不良レベルの判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、…、第 $n-2$ の不良レベルよりも下位の第 $n-1$ の不良レベルの判定基準と、前記第 $n-1$ の不良レベルの判定セグメント内に存在する不良ビット数とを比較し、この不良ビット数が前記判定基準以上である場合に前記判定セグメント毎に不良数をカウントし、第 $n-1$ の不良レベルよりも下位の第 n の不良レベルのセグメント内に存在する不良ビット数を、この第 n の不良レベルの不良数とするものであってもよい。

【0020】また、前記不良救済の可否を判断する工程は、 n 個の不良レベル毎に不良救済の可否を判断するものであり、前記第1の不良レベルの不良数と、この第1の不良レベルの救済処理単位内における置換可能なスペア数とを比較し、前記不良数が前記スペア数を超えている場合は救済不可と判断し、前記不良数が前記スペア数以下である場合は第1の不良レベルで救済可と判断し、前記第1の不良レベルで救済可と判断した場合、前記第2の不良レベルの不良数と、この第2の不良レベルの救済処理単位内における置換可能なスペア数とを比較し、前記不良数が前記スペア数を超えている場合は救済不可と判断し、前記不良数が前記スペア数以下である場合は第2の不良レベルで救済可と判断し、…、前記第 $n-2$ の不良レベルで救済可と判断した場合、前記第 $n-1$ の不良レベルの不良数と、この第 $n-1$ の不良レベルの救済処理単位内における置換可能なスペア数とを比較し、前記不良数が前記スペア数を超えている場合は救済不可と判断し、前記不良数が前記スペア数以下である場合は第 $n-1$ の不良レベルで救済可と判断し、前記第 $n-1$ の不良レベルの不良数と、この第 n の不良レベルの救済処理単位内における置換可能なスペア数とを比較し、前記不良数が前記スペア数を超えている場合は救済不可と判断し、前記不良数が前記スペア数以下である場合は第 n の不良レベルで救済可と判断するものであってもよい。

【0021】本発明の不良救済判定装置は、冗長回路を有する半導体記憶装置に対して複数の試験項目毎に試験を行い、不良救済の可否を判定する装置であって、前記

6

半導体記憶装置に試験項目毎に試験を行い、所定の大きさを有するセグメント毎の不良ビット数を測定して不良情報を作成するテストと、前記テストから不良救済判定装置に前記不良情報を与えられ、前記不良情報に基づき、複数の不良レベルに存在するそれぞれの不良数を検出して不良レベルの分類を行い、各々の不良レベルの不良数と、前記冗長回路に置換可能なスペア数とを比較して、不良レベル毎に不良救済の可否を判断する不良救済判定装置とを備えたことを特徴とする。

【0022】前記不良救済判定装置とは独立して、前記不良情報に基づき不良箇所を前記冗長回路に置き換えるアドレス情報を生成する不良解析装置をさらに備えることもできる。

【0023】

【発明の実施の形態】以下、本発明の一実施の形態について図面を参照して説明する。

【0024】本実施例による半導体記憶装置の不良判定処理方法は、図1に示されるような手順で不良判定を行う。

【0025】上述したように、従来はテストを用いた試験行程と、得られたデータに基づいてRAを用いて救済可能か否かの判定を行う解析工程とを、マージンテストにおける試験項目(B)～(F)毎に並行して行っていた。このため、RAによる救済判定処理に時間がかかる間、次の試験項目に移行するまでテストが待機しなければならず、処理効率の低下を招いていた。

【0026】これに対し、本実施の形態では、テストによりベーシックテスト(A)を行ってRAにより救済判定を行った後、テストにより他の全ての試験項目(B)～(F)におけるマージンテスト(B)～(F)を全い、RAにより試験項目(F)のみについて救済判定を行って、ヒューズデータを作成する。その後、テスト及びRAと独立して新たに設けたFBC(Fail Bit Counter)システムにより試験項目(A)～(F)に対する不良分類処理、及び試験項目(B)～(E)に対する救済判定処理を行う。即ち、試験項目(B)～(F)毎にテストによるマージンテストとRAによる救済判定処理とを並行して行うのではなく、マージンテストを全て完了した後、FBCシステムにより一括して不良分類及び救済判定処理を行う。このため、RAの処理が終了するまでテストが待機する必要がなく、処理効率が向上する。

【0027】まず、テストにより半導体記憶装置に対してDCテストを行う。さらに、例えば読み書き等の基本的な機能を調べるベーシックテスト(A)を行い、不良情報(A)を生成する。

【0028】この不良情報(A)をRAに送信して解析し、救済判定を行う。この救済判定を行っている間、テストによりマージンテスト(B)が行われる。救済判定の解析結果(A)が得られると、RAからテストへこの情報が送信される。また、テストにおいてマージンテ

7

トによる不良情報(B)が得られて蓄積される。

【0029】次に、テストによりマージンテスト(C)～(F)が行われ、順にFBCデータ(B)～(F)が得られ、テストに蓄積される。また、テストがマージンテスト(F)を終えた後、不良情報(F)を生成してRAに送信する。RAは、不良情報(F)に対してのみ救済判定を行い、ヒューズデータを含む救済判定結果

(F)を生成してテストに送信する。ヒューズデータを含む救済判定結果(F)を得るためには、不良箇所を特定するためのアドレス情報が必要である。後述するよう
10 に、FBCデータは圧縮されてアドレス情報が含まれないので、圧縮前の不良情報(F)を用いてヒューズデータを作成する。

【0030】また、テストはベーシックテスト(A)及びマージンテスト(B)～(F)毎に得られたFBCデータ(A)～(F)を得てデータファイル化して蓄積し、FBCシステムに送信する。試験項目(A)～

(F)のうち、項目(A)及び(F)はRAにおいて救済判定が既に行われているので、不良分類のみを行う。試験項目(B)～(E)に対しては、救済判定及び不良
20 分類処理を行う。ここで、不良分類とは、例えばブロック不良、ロー不良、カラム不良、ビット不良といった各不良レベル毎の不良数を検出する処理をいう。

【0031】ここで、不良情報(A)及び(F)は、圧縮処理を施す前の情報であって、アドレス情報を含んでいる。FBCデータ(B)～(F)は、各試験項目

(A)～(F)毎の試験により得られた不良情報を、後述するようにセグメント内の不良数を示す情報に圧縮したものであり、アドレス情報は含んでいない。

【0032】上記手順により得られた不良分類及び救済
30 判定処理により、試験項目(A)～(F)のうちいずれの項目で多く不良が検出されたかを調べることができる。不良が多く検出された試験項目から、半導体記憶装置の製造工程においていずれの工程で不良が多く発生したかを解析し、工程の改善を行い歩留まりの向上に寄与することができる。

【0033】また、上記実施の形態では1つの半導体記憶装置に対して1つのテストを用いて試験を行っている。しかし、複数の半導体記憶装置にそれぞれテストを
40 配置し、複数のテストにおいて得られたFBCデータを1つのFBCシステムで不良解析処理を行ってもよい。

【0034】複数の半導体記憶装置に対して試験を行う場合は、ある試験項目において全ての被試験対象が救済不可であることが判明した時点で、全ての試験を終了することも可能である。このような手法を用いることで、試験時間を短縮することができる。

【0035】次に、テストがFBCデータ(A)～(F)を収集する際の処理の手順について説明する。

【0036】ベーシックテスト(A)及びマージンテスト(B)～(F)を行った後、検出した不良ビット数を
50

8

カウントする。RAによる救済判定処理は各試験項目毎に行うのではなく、全てのテスト(A)～(F)が終了した後に一括して行う。但し、不良が多く発生し易いクリティカルな試験項目に関しては、当該試験項目に対してのみ従来のようにリアルタイムでRAにより救済判定処理を行うこともできる。例えば、図1に示されたように、試験項目(A)及び(F)がクリティカルであるとすると、このベーシックテスト(A)及びマージンテスト(F)を行った後、圧縮前の不良情報(A)及び
(F)をRAに送信して不良救済判定を行ってもよい。これにより、この試験項目で不良が多く発生し救済不可であると判定した場合に、この段階で試験を終了することで、処理効率を向上させることができる。

【0037】クリティカルでない他の試験項目(B)～(F)においては、RAにおいて一括して不良判定処理を行うために、セグメントを単位として不良数をカウントした圧縮情報を生成する。ここで、セグメントは半導体記憶装置における被試験対象領域の平面上の構成と、適切な救済単位の大きさを考慮して設定する。1つのセグメントの面積を大きくして全体のセグメント数を少なくすると、処理効率は高くなる。しかし、FBCシステムにおいて行う不良分類が困難になる。そこで、処理効率と不良分類の容易さとを比較考量してセグメントの最適なサイズを設定する。

【0038】例えば、図3に示されたように、ロードレス方向はビット線のサイズ、カラムアドレス方向はカラム救済単位とする。ここで、ワード線は負荷容量を低減させるために、ワード線0が複数本設けられてそれぞれのメモリセルアレイのブロックに接続されており、ワード線0がワード線1に接続された構成となっている。

【0039】図4に、幾つかの不良が存在する不良モデルを示す。テストが、セグメントを単位として不良ビット数をカウントし、セグメント毎の不良ビット数を示した図5のようなFBCデータマトリクスを生成する。FBCデータマトリクスはセグメント毎の不良数を示すものであるため、図4に示された不良モデルから図5に示されたFBCデータマトリクスを生成する過程で、アドレス情報が消滅する。

【0040】FBCデータマトリクスにおける不良カウント数は、試験項目を経る毎に蓄積され増加していく。例えば、試験項目(A)のテストで図5に示されたようなFBCデータマトリクスが得られ、試験項目(B)のテストまで行って図6に示されたようなFBCデータマトリクスが得られたとする。試験項目(B)のみのマージンテストでカウント数は、図5のFBCデータマトリクスから図6のFBCデータマトリクスへの不良数の増加分に相当する。この増加分の不良数のみを示したFBCデータマトリクスは、図7に示されたようであり、データ量が減少する。

【0041】このようにして、各試験項目毎に得られた

9

不良ビット数を各セグメント単位でカウントし、FBCデータとして蓄積していく。このようなFBCデータを用いることにより、テストと非同期で救済判定処理を行うことが可能であり、高額なテストを待機させる時間を大幅に減少させることができる。

【0042】次に、本実施の形態における救済判定処理の手順について図2を用いて説明する。上述したように、FBCデータを用いてFBCシステムにより集中的に救済判定処理を行う。FBCシステムでは、テストから送信されてくるFBCデータを常時受信することができ、常駐プロセス（以降、FBCプロセスと称する）を走らせておく必要がある。

【0043】そこで、ステップS100としてFBCプロセスを起動する。FBCプロセスは、常時FBCファイルの受信を監視する。ステップS101としてFBCファイルを受信し、ステップS102においてFBCファイルが存在すると判断すると、FBCデータを読取って不良分類／救済判定の処理プロセス（以降、RAシミュレータと称する）へ受け渡す。

【0044】ここで、FBCデータには、例えばヘッダ情報としてウェーハ情報、チップ情報、試験項目情報等が書きこまれている。これらの情報を用いることで、RAはRAシミュレータを起動する必要があるか否かを判断することができる。

【0045】先ず、ステップS104としてウェーハ情報を読み取り、次にステップS106としてチップ情報を読み取り、救済対象となっているチップであるか否かを判断し、対象となっている場合は次のステップS110へ移行し、対象となっていない場合は上記ステップS106へ戻って次のチップに対してステップS108の判断を行う。

【0046】ステップS110として、ヘッダ情報としての試験項目情報を読み取り、ステップS112としてFBCデータを読み取る。ヘッダ情報からRAシミュレータの起動が要求された場合には、ステップS114としてRAシミュレータが起動される。

【0047】RAシミュレータが起動すると、FBCデータが用いられ救済判定が行われる。ステップS116として救済が可能であるか否かが判断され、救済可能である場合は上記ステップS110へ戻って次の試験項目の情報が読み取られ、FBCデータの読み取り及びRAシミュレータの起動が行われ、救済判定が行われる。救済不可である場合はステップS118へ移行し、全チップに対して処理が終了したか否かが判断される。終了していない場合は、ステップS106へ戻って次のチップに対する処理を開始し、終了した場合はステップS120として不良カテゴリがデータベースに登録される。不良カテゴリには、最終的に半導体記憶装置が良品と判断されたか否か、また各不良レベルにおける不良数を示す情報が含まれる。

10

【0048】図8に、RAシミュレータによる処理の手順を示す。RAシミュレータには、大別するとステップS200としての不良分類処理と、ステップS202としての救済判定処理の2つの処理系が存在する。

【0049】ステップS200における不良分類処理では、不良レベルの推定を行うもので、例えばブロック不良、カラム選択線不良、ワード線不良、ビット線不良、セル不良等の幾つかの不良レベル毎の不良数を検出する。

【0050】ステップS202における救済判定処理では、分類された各不良レベルの不良数と、冗長回路に置換可能なスペア数とを救済判定処理単位毎に比較し救済の可否を判定する。

【0051】ステップS204で救済不可であると判定した場合は、ステップS210として救済不可であるというステータスを設定する。救済可能である場合はステップS206において全ての救済判定処理単位数が終了したか否かを判断し、終了した場合はステップS208として救済可というステータスを設定する。

【0052】このようにして設定された救済可又は不可を示すステータス情報が、RAシミュレータからの出力結果としてFBCプロセスに返されて、対象チップの不良カテゴリとして登録される。

【0053】上述した処理により、従来テスト及びRAにより判定し作成していた不良カテゴリ情報を、テストから切り離したオフラインによりFBCシステムで作成することができるので、テストを待機させることがなく処理効率が向上する。

【0054】次に、（１）不良分類処理、（２）救済判定処理の2つの処理に大別され、それぞれについて、以下に詳細に説明する。

【0055】（１）不良分類処理

半導体記憶装置は、一般に図9に示されたように不良レベルを設定することができる。基本的な不良分類のステップは、大きい不良から小さい不良に向かって行っていく。例えば、ブロック、カラム選択線、ワード線1、ビット線、ワード線0、セルの順番に分類していく。

【0056】いずれの不良レベルに分類すべきかという判定基準は、対象とする半導体記憶装置の構造により異なり、判定セグメント内に存在する不良ビット数が当該判定基準を超えているか否かにより判定を行う。

【0057】図9に示された判定基準を用いて不良分類処理の手順を、図10及び図11のフローチャートに示す。RAシミュレータを起動し、図8に示されたステップS200として不良分類処理を開始する。全判定セグメントに対し、ステップS300としてブロック不良の有無を調べる。ブロック不良の判定基準BlkLimitは、例えば図9に示されたように、ブロックサイズBlkSize

（判定セグメント数、ここでは4）に、判定精度を高めるための係数fBlk（ $0 \leq fBlk \leq 1$ ）を乗算したものをを用

11

いる。この判定基準 $BlkLimit$ と、判定セグメント内の不良ビット数 $FB C b$ とを比較する。そして、不良ビット数 $FB C b$ が判定基準 $BlkLimit$ を超えている場合は、ステップ $S 3 0 2$ として不良ビット数 $FB C b$ からブロックサイズ $BlkSize$ を差し引き、ブロックカウント数 $BlkCount$ をカウントアップする。

【0058】ステップ $S 3 0 4$ として、全判定セグメントに対してブロック不良か否かの判断を行ったと判断した場合、ステップ $S 3 0 6$ としてカラム選択線の不良に関して判定を行う。カラム選択線不良の判定基準 $M2Limit$ は、カラム選択線サイズ $M2Size$ に判定精度向上用の係数 $fM2$ を乗算したものをを用いる。この判定基準 $M2Limit$ と、判定セグメント内の不良ビット数 $FB C m 2$ とを比較する。そして、不良ビット数 $FB C m 2$ が判定基準 $M2Limit$ を超えている場合は、ステップ $S 3 0 8$ として不良ビット数 $FB C m 2$ からカラム選択線サイズ $M2Size$ を差し引き、カラム選択線カウント数 $M2Count$ をカウントアップする。

【0059】ステップ $S 3 1 0$ として、全セグメントに対してカラム選択線不良か否かの判断を行ったと判断した場合、ステップ $S 3 1 2$ としてワード線の不良に関して判定を行う。ワード線不良の判定基準 $MLimit$ は、ワード線サイズ $MSize$ に判定精度向上用の係数 $fM1$ を乗算したものをを用いる。この判定基準 $MLimit$ と、判定セグメント内の不良ビット数 $FB C m 1$ とを比較する。そして、不良ビット数 $FB C m 1$ が判定基準 $MLimit$ を超えている場合は、ステップ $S 3 1 4$ として不良ビット数 $FB C m 1$ からワード選択線サイズ $MSize$ を差し引き、ワード線カウント数 $M1Count$ をカウントアップする。

【0060】ステップ $S 3 1 6$ として、全セグメントに対してワード線不良か否かの判断を行ったと判断した場合、ステップ $S 3 1 8$ としてビット線不良に関して判定を行う。ビット線不良の判定基準 $MOLimit$ は、ビット線サイズ $MOSize$ に判定精度向上用の係数 $fM0$ を乗算したものをを用いる。この判定基準 $MOLimit$ と、判定セグメント内の不良ビット数 $FB C$ とを比較する。そして、不良ビット数 $FB C$ が判定基準 $MOLimit$ を超えている場合は、ステップ $S 3 2 0$ として不良ビット数 $FB C$ からビット線サイズ $MOSize$ を差し引き、ビット線カウント数 $MOCOUNT$ をカウントアップする。

【0061】ステップ $S 3 2 2$ として、全セグメントに対してビット線不良か否かの判断を行ったと判断した場合、ステップ $S 3 2 4$ としてワード線0の不良に関して判定を行う。ワード線0不良の判定基準 $GCLimit$ は、ワード線0サイズ $GCSize$ に判定精度向上用の係数 fGC を乗算したものをを用いる。この判定基準 $GCLimit$ と、判定セグメント内の不良ビット数 $FB C$ とを比較する。そして、不良ビット数 $FB C$ が判定基準 $GCLimit$ を超えている場合は、ステップ $S 3 2 6$ として不良ビット数 $FB C$ からワード線0サイズ $GCSize$ を差し引き、ワード線0カ

12

ウント数 $GCCOUNT$ をカウントアップする。

【0062】ステップ $S 3 2 8$ として、全セグメントに対してワード線0不良か否かの判断を行ったと判断した場合、ステップ $S 3 3 0$ へ移行する。最終的に上記判定基準のいずれにも該当せずに残った不良ビット数が、セル不良数 $CellCount$ としてカウントされる。ステップ $S 3 3 2$ において全てのセグメントに対する分類が終了したか否かを判断し、終了した場合は不良分類処理を終了する。

【0063】(2) 救済判定処理

救済判定は、ブロック救済、ロー救済、カラム救済、ビット系救済のそれぞれが可能であるか否かを判定するものである。それぞれの判定基準は図12に示されるようであり、救済判定の処理手順は図13のフローチャートに示されるようである。救済判定処理は、ブロック救済、ロー救済もしくはカラム救済、ビット系救済の順に行われる。それぞれの救済判定は、各救済処理単位内における不良数の総和が、冗長回路に置換可能なスペア数、即ち判定基準を超えているか否かにより行う。

【0064】先ず、ステップ $S 4 0 0$ としてブロック救済が可能か否かを判定する。本実施の形態では、ブロック救済に対応した冗長構成を備えておらず、ブロック救済判定基準は0である。このため、ブロック不良数 $BlkCount$ が1以上であると救済不可となり、ステップ $S 4 1 8$ として救済不可ステータスを設定して処理を終了する。ブロック不良数 $BlkCount$ が0である場合のみ、次のステップ $S 4 0 2$ へ移行し、全救済単位における処理が可能である場合ステップ $S 4 0 4$ へ移行する。

【0065】ロー救済とカラム救済は並列関係にあり、いずれを先に行ってもよいが、ここではステップ $S 4 0 4$ としてロー救済の判定処理を行う。ロー救済単位内のロウ不良総数 $RowCount$ が判定基準 $RowSp$ を超えていた場合、ステップ $S 4 1 8$ として救済不可ステータスを設定し処理を終了する。

【0066】ステップ $S 4 0 6$ において全救済単位においてロー救済が可能であると判断すると、ステップ $S 4 0 8$ へ移行してカラム救済処理を行う。カラム救済単位内のカラム不良総数 $ColCount$ が判定基準 $ColSp$ を超えていた場合、救済不可ステータスを設定し処理は終了する。

【0067】ステップ $S 4 1 0$ として全救済単位に対してカラム救済可であると判断した場合、ステップ $S 4 1 2$ としてビット系救済を行う。ビット系不良数 $CellCount$ が、残っているスペア数、即ちロー救済及びカラム救済後に残っているスペア数 $(RowSp + ColSp) - (RowCount + ColCount)$ を超えていた場合、ステップ $S 4 1 8$ として救済不可ステータスを設定して処理を終了する。ビット系救済可である場合は、ステップ $S 4 1 6$ として救済可ステータスを設定して全処理を終了する。

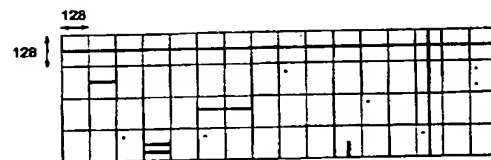
【0068】上述した手順により得られた不良カテゴリ

14

【図１】本発明の一実施の形態による半導体記憶装置の不良判定処理方法を従来の方と対比して示した説明図。

【図13】同救済判定処理の手順を示したフローチャート。

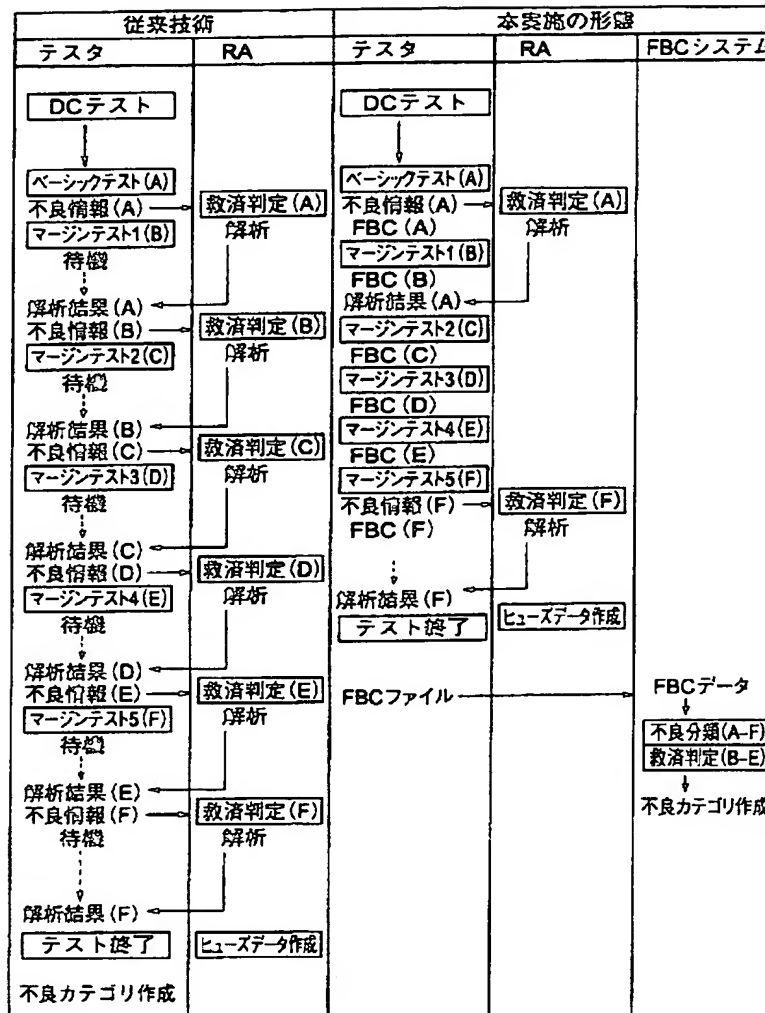
【図 4】



【图6】

128	128	128	128	128	128	129	129	128	128	128	128	128	255	128	128
	128						1					2	128		2
	128			1	128	128					1		128		
		1	256		1					67			129		

【図1】



【図7】

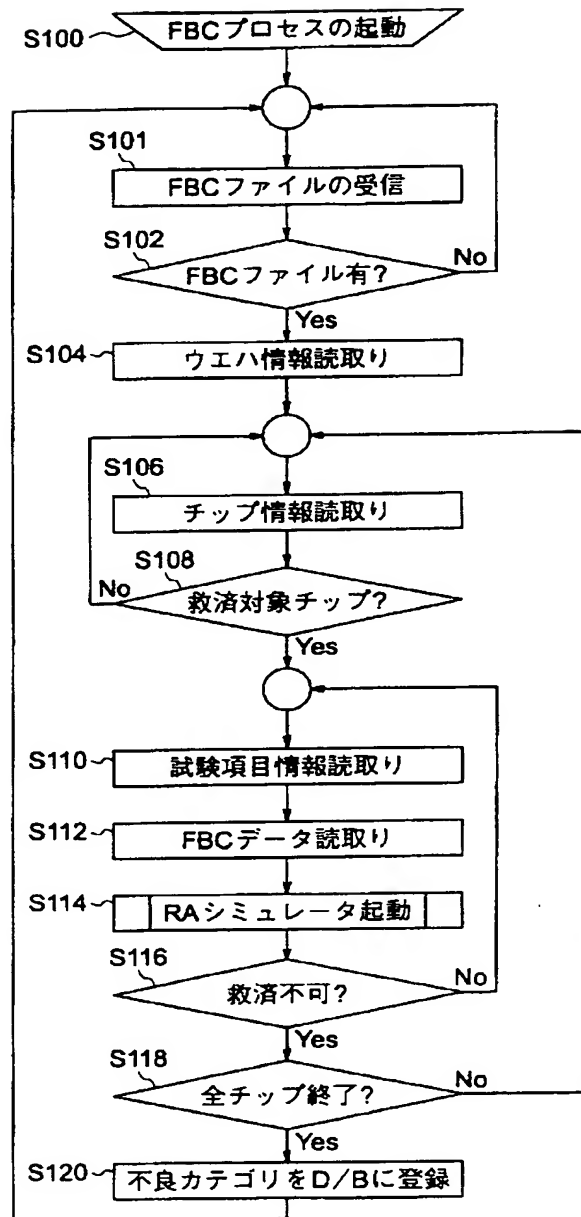
				1										
					1			2						
128		1												
						3								

【図12】

	救済	不良数	判定凸凹
1	ブロック救済	BlkCount	0
2	ロウ救済	RowCount = M1Count + GCCount	RowSp
3	カラム救済	ColCount = M2Count + 16 + M3Count	ColSp
4	ビット系救済	CellCount	(RowCount + ColSp) - (RowCount + ColCount)

注1) 判定凸凹のRowSp, ColSpは、各救済単位内で使用可能なRow/Columnスペースを示す。

【図2】



【図9】

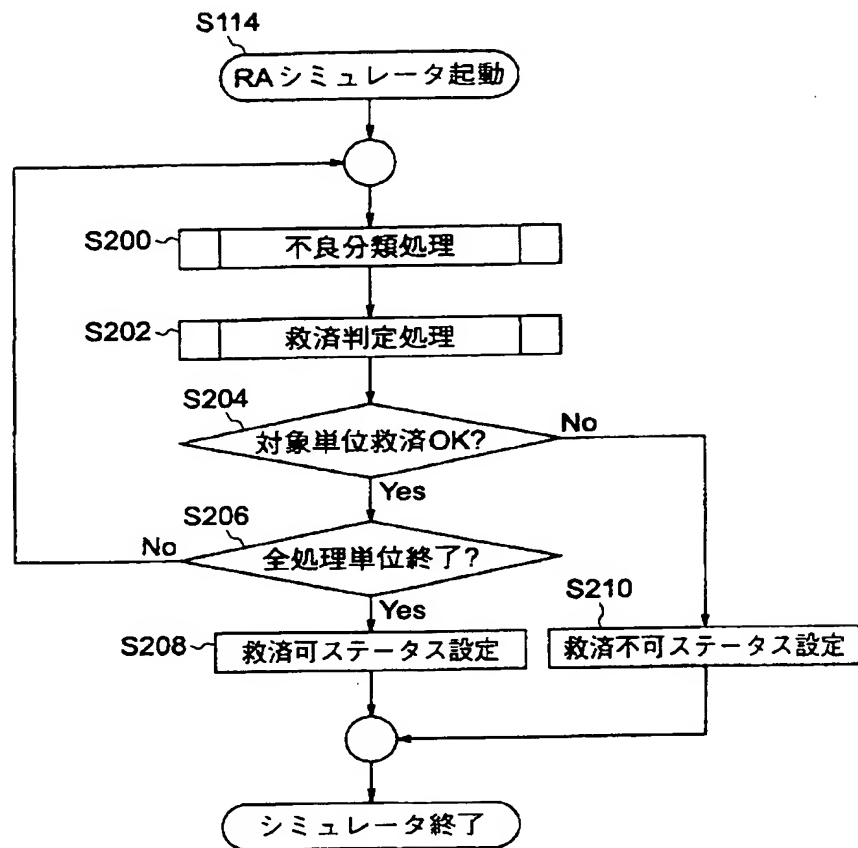
	不良レベル	判定セグメント数	不良ビット数	判定基準	判定結果
1	ブロック	4 (ワード線方向)	FBCb	$BkLimit = BkSize * fBk$	BlockCount (Block)
2	カラム選択線	16 (ビット線方向)	FBCm2	$M2Limit = M2Size * fM2$	M2Count (Column)
3	ワード線1	4 (ワード線方向)	FBCm1	$M1Limit = M1Size * fM1$	M1Count (Row)
4	ビット線	1	FBC	$M0Limit = M0Size * fM0$	M0Count (Column)
5	ワード線0	1	FBC	$GCLimit = GCSize * fGC$	GCCount (Row)
6	セル	1	FBC	上記基準値外	CellCount (bit系)

注1) 不良数のFBCb, FBCm2, FBCm1は、判定セグメント内の総和を示している。

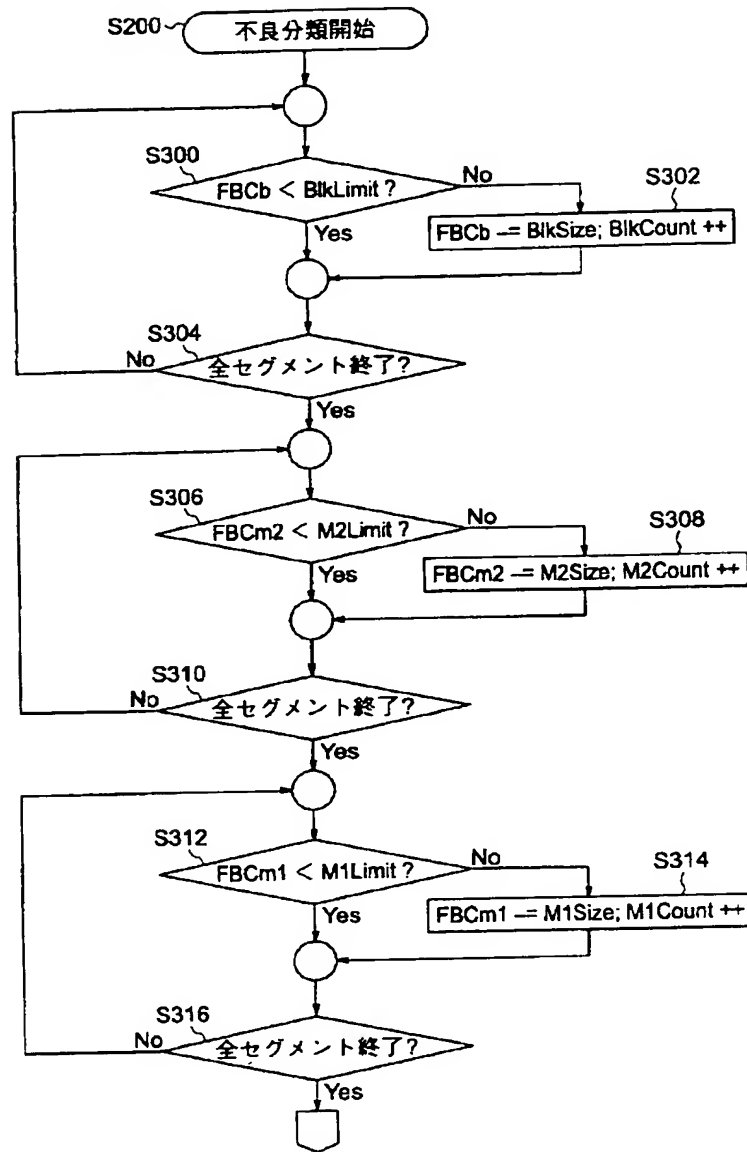
注2) 判定基準のfBk, fM2, fM1, fM0, fGCは、判定精度を保つための係数であり、それぞれ任意に設定できる。

注3) 判定結果の**Countは、各不良レベルの総数を表し、後の救済判定で使われる。

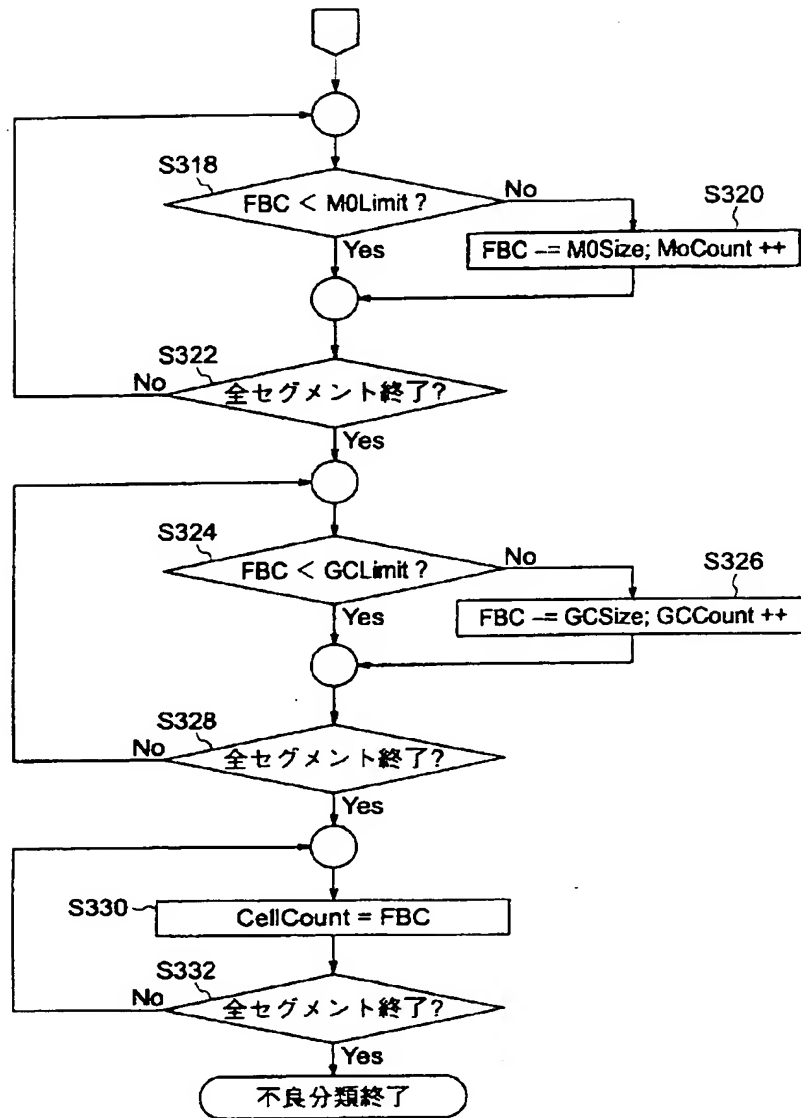
【図8】



【図10】



【図 11】



【図13】

